

## CLAIMS

1. A method, comprising:

receiving a request to add a new filter to a filter table stored in an inverse query engine cache;

adding the new filter to the filter table;

maintaining the inverse query engine cache at or below a maximum cache size; and

wherein the inverse query engine cache is used exclusively by an inverse query engine to store filters associated therewith.

2. The method as recited in claim 1, further comprising maintaining the size of the inverse query engine cache between an optimal cache size and the maximum cache size.

3. The method as recited in claim 1, wherein the maintaining further comprises:

determining if the addition of the new filter to the new filter table increases the cache size above the maximum cache size; and

removing one or more filters from the filter table if the addition of the new filter causes the cache size to exceed the maximum cache size.

4. The method as recited in claim 3, wherein the determining step further comprises:

determining a relative size of the new filter;

assigning a filter weight to the new filter based on the relative filter size;

1 deriving a cache weight by summing filter weights of all filters in the filter  
2 table including the new filter; and

3 comparing the cache weight to the maximum cache size.

4  
5 5. The method as recited in claim 1, wherein the maintaining further  
6 comprises:

7 identifying a weight associated with the new filter;

8 adding the weight associated with the new filter to a cache weight that is  
9 the sum of filter weights of filters in the filter table, each filter having a filter  
10 weight; and

11 comparing the cache weight to the maximum cache size.

12  
13 6. The method as recited in claim 1, further comprising identifying one  
14 or more expired filters in the filter table; and wherein the maintaining the inverse  
15 query engine cache further comprises removing at least one of the identified  
16 expired filters.

17  
18 7. The method as recited in claim 1, further comprising a least recently  
19 used filter in the filter table; and wherein the maintaining the inverse query engine  
20 cache further comprises removing the least recently used filter from the filter table  
21 when a size of the inverse query engine cache reaches the maximum cache size.

1           **8.**    A system, comprising:  
2           an inverse query engine configured to test an input against a collection of  
3 filters;  
4           cache associated with the inverse query engine;  
5           a filter table stored in the cache and containing multiple filters; and  
6           a maintainer configured to maintain a size of the filter table within definite  
7 cache bounds.

8  
9           **9.**    The system as recited in claim 8, further comprising an expiration  
10 module configured to remove expired filters from the filter table.

11  
12           **10.**   The system as recited in claim 9, wherein the maintainer is further  
13 configured to invoke the expiration module when a new filter is added to the filter  
14 table.

15  
16           **11.**   The system as recited in claim 8, further comprising a trim module  
17 configured to remove one or more filters from the filter table when the cache  
18 reaches a maximum cache size.

19  
20           **12.**   The system as recited in claim 11, wherein the trim module is  
21 further configured to remove the one or more filters from the filter table until the  
22 cache is reduced to an optimal cache size.

1           **13.**     The system as recited in claim 11, wherein the trim module is  
2 further configured to determine if a permanent flag in a filter is set and, if the  
3 permanent flag is set, to leave the filter in the filter table.

4  
5           **14.**     The system as recited in claim 11, wherein:  
6 a filter weight is associated with each filter in the filter table;  
7 the cache size further comprises a cache weight that is a sum of all filter  
8 weights in the filter table;  
9 the maximum cache size further comprises a maximum cache weight; and  
10 wherein the trim module is further configured to deduct a filter weight from  
11 the cache weight when a filter associated with the filter weight is removed from  
12 the filter table.

13  
14           **15.**     The system as recited in claim 14, further comprising a cache weight  
15 module configured to assign a filter weight to each filter added to the filter table,  
16 each filter weight identifying a relative size of a filter with regard to other filters in  
17 the filter table

18  
19           **16.**     The system as recited in claim 11, further comprising:  
20 a most recently used list configured to track usage of filter table filters  
21 according to when filters are used, and indicating that a filter has been most  
22 recently used when the filter is the latest filter to match an input; and  
23 wherein the trim module is further configured to remove the one or more  
24 filters from the filter table based on which filters have been least recently used.

1           **17.**     One or more computer-readable media storing computer-executable  
2 instructions that, when executed on a computer, perform the following steps:  
3           receiving a request to add a new query to an inverse query engine cache  
4 that stores multiple queries, each query having a query size associated therewith;  
5           deriving a cache size that is a sum of query sizes of the queries stored in the  
6 inverse query engine;  
7           determining if the cache size is at greater than or equal to a maximum cache  
8 size;  
9           removing one or more queries from the inverse query engine cache if the  
10 cache size is greater than or equal to the maximum cache size;  
11          deducting the query size of each query removed from the cache size;  
12          adding the new query to the inverse query engine cache; and  
13          adding a new query size to the cache size, the new query size identifying a  
14 size of the new query added to the inverse query engine cache.

15  
16           **18.**     The one or more computer-readable media as recited in claim 17,  
17 wherein the removing step further comprises removing a query from the inverse  
18 query engine that has been used less recently than other queries stored in the  
19 inverse query engine cache.

20  
21           **19.**     The one or more computer-readable media as recited in claim 17,  
22 wherein a query size is an estimate of the actual size of the query.  
23  
24  
25

1           **20.**     The one or more computer-readable media as recited in claim 17,  
2 wherein the step of adding the new query size to the cache size is performed  
3 before determining if the cache size is greater than or equal to the maximum cache  
4 size.

5  
6           **21.**     The one or more computer-readable media as recited in claim 17,  
7 further comprising instructions to perform the following additional steps:  
8           determining if any queries stored in the inverse query engine cache have  
9 expired; and  
10          removing one or more expired queries from the inverse query engine cache.

11  
12          **22.**     The one or more computer-readable media as recited in claim 17,  
13 wherein the new query size is received with the new query.

14  
15          **23.**     The one or more computer-readable media as recited in claim 17,  
16 further comprising instructions to perform the additional step of determining the  
17 new query size.

18  
19          **24.**     The one or more computer-readable media as recited in claim 23,  
20 wherein the determining the new query size further comprises estimating the new  
21 query size.

1           **25.**     The one or more computer-readable media as recited in claim 17,  
2 further comprising the additional step of:  
3           determining if a query in the inverse query engine cache includes an  
4 indication that the query is a permanent query; and  
5           wherein the query is not removed from the inverse query engine cache if  
6 the query is a permanent query.

7  
8           **26.**     The one or more computer-readable media as recited in claim 17,  
9 wherein a query size is represented as a weight value that denotes the relative size  
10 of the query associated therewith with regard to other queries stored in the inverse  
11 query engine cache.

12  
13           **27.**     A method for maintaining an inverse query engine cache,  
14 comprising:  
15           determining when inverse query engine cache usage is approaching a cache  
16 usage capacity;  
17           removing one or more filters from the inverse query engine cache when the  
18 cache is approaching the cache capacity until the cache usage is reduced to an  
19 optimal cache usage;

20  
21           **28.**     The method as recited in claim 27, wherein the removing step  
22 further comprises removing at least one expired filter from the inverse query  
23 engine cache.

1           **29.**     The method as recited in claim 28, wherein an expired filter is a  
2 filter having an expiration time associated therewith that precedes a current time.

3  
4           **30.**     The method as recited in claim 28, wherein an expired filter is a  
5 filter that has been stored in the inverse query engine cache for a predefined period  
6 of time.

7  
8           **31.**     The method as recited in claim 27, wherein the removing step  
9 further comprises removing a least recently used filter from the inverse query  
10 engine cache.

11  
12           **32.**     An inverse query engine having an integrated cache.

13  
14           **33.**     The inverse query engine as recited in claim 32, wherein the cache  
15 is bound to a finite size.

16  
17           **34.**     The inverse query engine as recited in claim 32, wherein the cache  
18 is maintained within predefined limits by removing expired filters from a set of  
19 filters stored in the cache.

20  
21           **35.**     The inverse query engine as recited in claim 34 wherein an expired  
22 filter is a filter having an expiration time associated therewith than is earlier than a  
23 current time.



1           **36.**     The inverse query engine as recited in claim 34 wherein an expired  
2 filter is a filter that has been stored in the cache for at least a specified period of  
3 time.

4  
5           **37.**     The inverse query engine as recited in claim 32, wherein the cache  
6 is maintained within predefined limits by removing at least one filter from a group  
7 of filters stored in the cache that has been used less recently than other filters in  
8 the group of filters.

9  
10          **38.**     The inverse query engine as recited in claim 37, wherein the at least  
11 one filter is removed only if the filter does not have an indication associated  
12 therewith that identifies the filter as a permanent filter that is not to be removed  
13 from the cache.